eip ESCUELA INTERNACIONAL POSGRADOS

Máster Avanzado de Programación en Pythn para Hacking, BigData y Machine Learning

Programación Avanzada Python

LECCIÓN 02

Controladores de flujo y colecciones de datos

ÍNDICE

INTRODUCCIÓN

En este segunda lección vamos a estudiar los principales controladores de flujo que tenemos en Python. Además, aprenderemos nuevos tipos de datos más potentes que los estudiados en la lección 1 como son las tuplas, conjuntos y diccionarios. Por último, veremos como mostrar y pedir datos por pantalla. Intentaremos ver todo este contenido de una forma practica aunque acompañado con un mínimo de teoría.

OBJETIVOS

Al finalizar esta lección serás capaz de:

1. Utilizar los controladores de flujo.
2. Trabajar con tuplas, conjuntos y diccionarios
3. Diferenciar entre los diferentes tipos de datos
4. Mostrar y pedir datos por pantalla.

Controladores de flujo

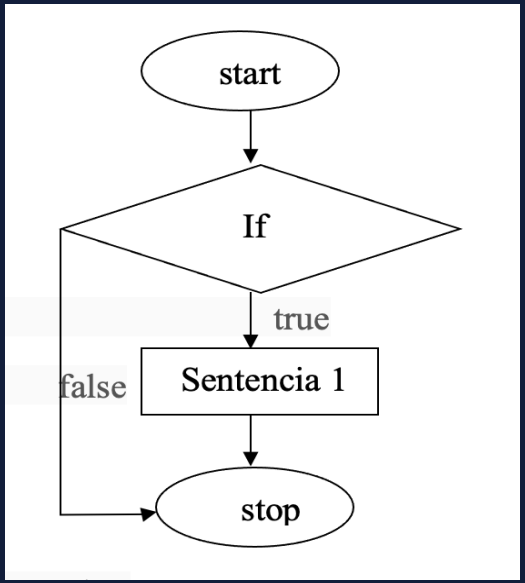
Los controladores de flujo determinan el flujo de ejecución de un programa basándose en algunas condiciones. Las condiciones deciden qué línea de código se ejecuta o no. Por lo tanto, todos los controladores de flujo son tomadores de decisiones

Sentencia IF

La sentencia If es el controlador de flujo más común y puede comprobar la condición con la palabra clave 'if'

if <condición>:

Sentencia 1



Sentencia IF

Veamos un ejemplo:

```
number = 5
if number < 10 :
  print("number is less than 10")
number is less than 10
```

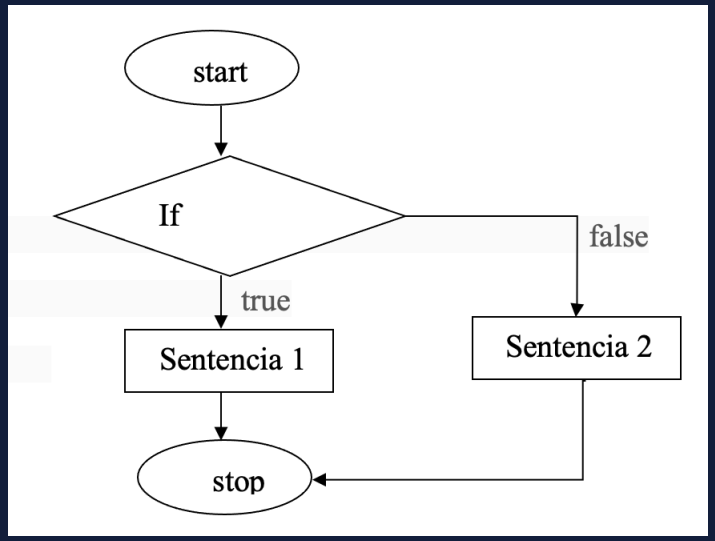
Los dos puntos después de la condición "if" son obligatorios y la declaración debe tener una sangría adecuada.

if-else

Cuando la condición es falsa, podemos utilizar la parte else si hay una sentencia que ejecutar.

Sintaxis:

```
If condición:
        sentencia 1
else:
        sentencia 2
```



if-else

Veamos un ejemplo:

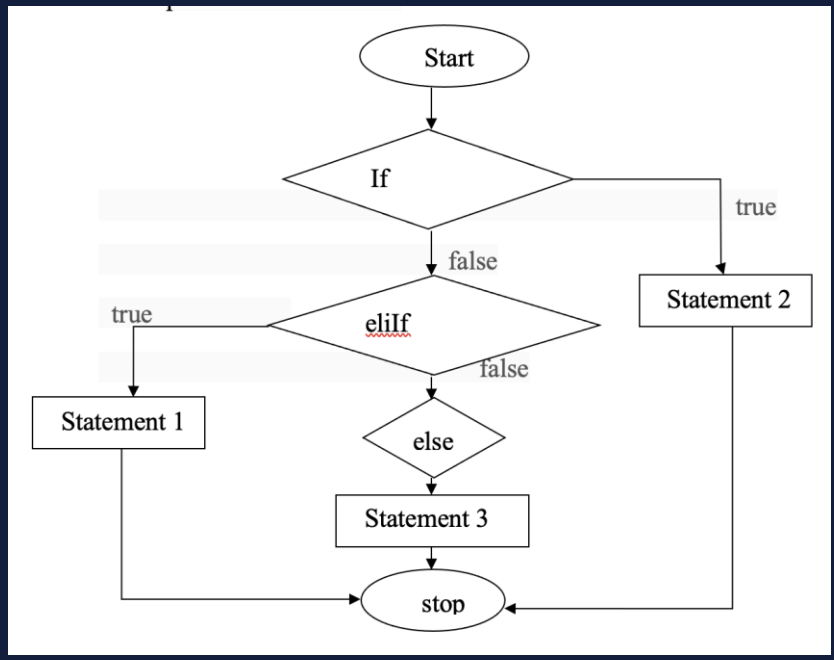
```
number = 12
if number < 10 :
  print("number is less than 10")
else:
  print("number is greater than 10")
```

```
number is greater than 10
```

Es necesario poner dos puntos después de la condición else y la sentencia debe tener la sangría adecuada.

If – elif – else

Si hay varias condiciones que comprobar, se puede utilizar la sentencia if- elif- else. 'elif' es una palabra clave utilizada en lugar de else if, que de nuevo comprueba una condición en el caso de que sea falsa la sentencia if.



If – elif – else

Sintaxis:

condición if:

declaración 1

condición elif:

declaración 2

elif .....

.....

else:

imprimir condición no válida

```
number = 12
if number < 10 :
  print("number is less than 10")
elif number > 15:
  print("number is greater than 15")
else:
  print("number is in between 10 and 15")
```

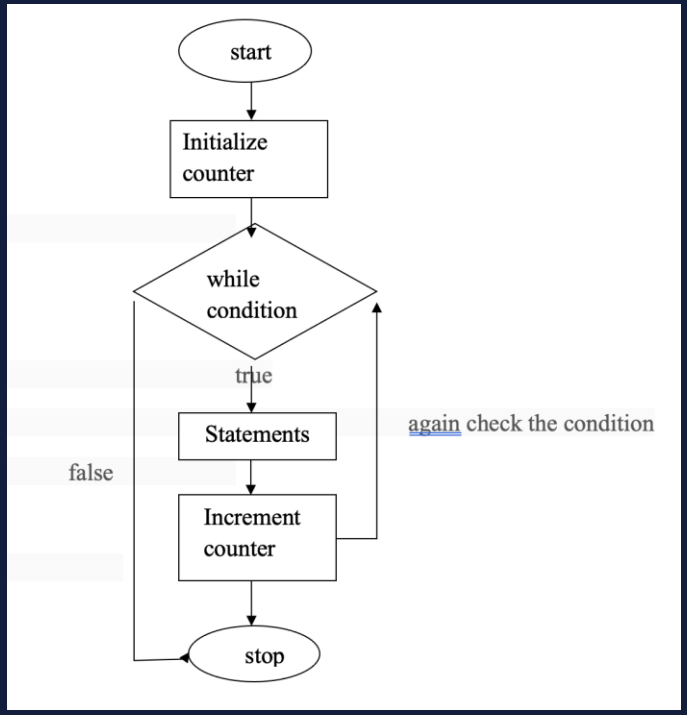
```
number is in between 10 and 15
```

Sentencia While

La sentencia while puede comprobar varias veces una condición particular para un conjunto de valores.

Sintaxis:

```
Variable_contador = 0
while  condition:
        statement1
        ..............
        Variable_contador += 1
```



Sentencia While

Veamos 2 ejemplos:

```
num = 1
while num <= 5:
  print(num)
  num += 1
```

```
1
2
3
4
5
```

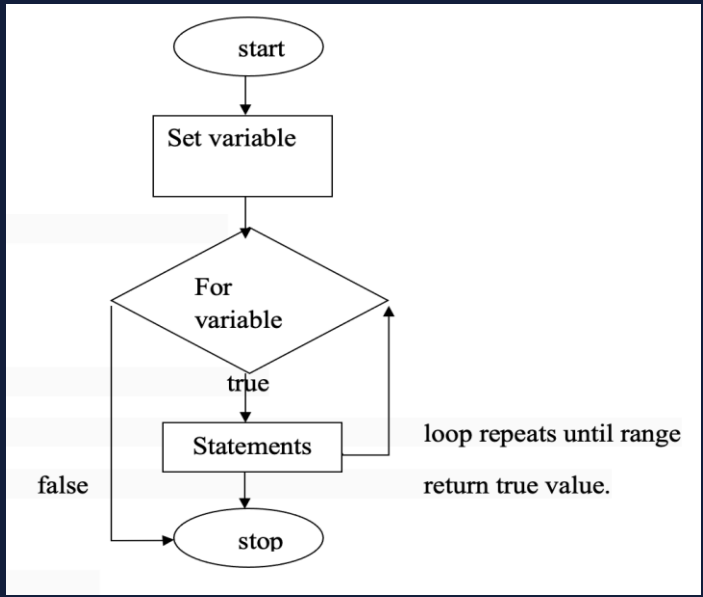
```
num = 1
while num <= 5:
  print(num)
```

```
1
1
1
1
1
1
1
```

Sentencia For

Los bucles while y for tienen el mismo propósito, se utilizan para ejecutar un conjunto de sentencias dada una condición. La mayor diferencia es que hay un incremento automático del bucle dentro del bucle for en lugar de la variable contador que debe ser incrementada manualmente en el bucle while.

```
for variable in range(n):
        statement 1
        ……..
```



Sentencia For

Veamos algunos ejemplos:

```
for value in range(5) :
  print(value)
```

```
0
1
2
3
4
```

```
for value in range(0,10,2) :
  print(value)
```

```
0
2
4
6
8
```

Bucle for anidado

```
for i in range(3):
  for j in range(2):
    print(i)
  print("\n")
```

Tuplas

La tupla es una estructura de datos que recoge un conjunto de elementos de diferentes tipos de datos. Una de las principales diferencias de las tuplas con respecto a otros tipos de datos es que, una vez creadas, no se pueden modificar. Los elementos de las tuplas se definen con paréntesis.

Sintaxis:

tuple_name = (item1. Item2… item n)

```
fruits =("apple", "orange", "banana")
fruits[0]
'apple'
```

Conversión entre lista y tupla

Una tupla se puede convertir en lista llamando a list (). El parámetro de la función es el nombre de la tupla.

```
new_list = list(fruits)
print(new_list, type(new_list))
```

```
['apple', 'orange', 'banana'] <class 'list'>
```

También el contenido de la lista se puede convertir en tupla.

```
new_tuple = tuple(new_list)
print(new_tuple, type(new_tuple))
```

```
('apple', 'orange', 'banana') <class 'tuple'>
```

Conjuntos

Los conjuntos es otra estructura de datos que recoge los elementos dentro de corchetes. La principal característica de los conjuntos es que son desordenados y la indexación no funciona.

Sintaxis:

nombre_conjunto = {elemento 1, elemento 2... elemento n}

```
country = {"India","USA","UAE"}
print(type(country))
<class 'set'>
```

Entrada de datos

La función *raw_input ()* se utiliza para leer un carácter o cadena y números desde el teclado. Siempre devuelve una cadena incluso si es un número.

```
number = raw_input("Enter a number:")
```

```
Enter a number:2
```

También es posible utilizar input()

```
number = input("Enter a number:")
```

```
Enter a number:2
```

Salida de datos

La función print() puede ser usada para imprimir los datos.

```
print(123)
```

```
123
```

```
print('number')
```

```
number
```

Otro método para la salida de los valores de las variables es usar el operador de módulo, %. Hay muchos formatos para diferentes tipos de datos que pueden ser dados con el operador modulo.

Algunas cadenas formateadas son

%d - int

%s - cadena

%f - float

```
num = 12
print('number: %d' % num)
```

```
number: 12
```

CONCLUSIONES

1. Las **listas**, **tuplas**, **diccionarios** y **conjuntos** (**set**) son estructuras que permiten trabajar con colecciones de datos.

2. En Python tenemos 3 principales controladores de flujo: *if, for y while*

3. Mediante las funciones **print** y **input** podemos mostrar y pedir datos por pantalla.

MUCHAS GRACIAS POR SU ATENCIÓN

eip ESCUELA INTERNACIONAL POSGRADOS

rsanchezi@grupomainjobs.com

Rubén Sánchez Iruela
linkedin.com/in/ruben-sanchez-iruela-8156799a

twitter.com/eiposgrados

facebook.com/eiposgrados

instagram.com/eiposgrados